IN THE CLAIMS

1. (**Currently Amended**) A method of accessing, by a client, one or more files residing in a server comprising:

requesting, by the client, downloading of a selected file residing in the server, the selected file needing access, either directly or indirectly, to at least one associated file in order to be used, the selected file including instructions to access the at least one associated file, the selected file further being associated with at least one profile**, wherein the profile identifies** ~~identifying, for download, only~~ the at least one associated file **by a globally unique identifier, wherein the globally unique identifier remains constant when the at least one associated file is renamed or relocated in the server**;

in response to requesting downloading of the selected file, initiating downloading of the selected file and automatically determining, at the client, the identity of and initiating downloading of the at least one associated file of the selected file to occur at substantially the same time as the selected file; ~~and~~

searching, by the client, for the at least one associated file using the globally unique identifier in response to a failure to find the at least one associated file in the server; and

initiating storing, in a memory associated with the client, of the selected file and the at least one associated file under respective local identifiers.

2. (Original) The method of Claim 1, and further comprising maintaining, by a document manager residing in the server, respective profiles of the one or more files.

3. (Cancelled)

4. (Original) The method of Claim 3, wherein the profile identifies the at least one associated file by the Uniform Resource Locator.

5. (Previously Presented) The method of Claim 1, wherein automatically determining the identity of and initiating downloading of the at least one associated file

comprises examining a profile of the selected file, the profile identifying the at least one associated file.

6. (Original) The method of Claim 1, and further comprising maintaining a respective status file for each of the selected file and the at least one associated file, each status file indicating one or more properties of the respective selected file and the at least one associated file.

7. (Original) The method of Claim 6, wherein the status file is a cookie file.

8. (Original) The method of Claim 6, wherein the status file consists solely of a timestamp indicative of a time of download.

9. (Original) The method of Claim 6, wherein the status file comprises a timestamp indicative of a time of download, a check out status, and respective identities of the at least one associated file.

10. (Original) The method of Claim 1, wherein the memory associated with the client is a root of a cache, the root identified by a root directory identifier.

11. (Original) The method of Claim 10, wherein each of the respective local identifiers comprises the root directory identifier.

12. (Original) The method of Claim 1, and further comprising:

generating, by the client, the one or more files for uploading to the server;

generating, by the client, a profile associated with each of the one or more files; and

uploading, by the client, the profile and the each of the one or more files to the server.

13. (**Currently Amended**) A method of accessing, by a client, one or more files managed by a document manager residing in a server, the method comprising:

requesting, by the client, downloading of a selected file residing in the server, the selected file needing access, either directly or indirectly, to at least one associated file in order to be used, the selected file further being associated with at least one profile**, wherein the profile identifies** ~~identifying, for download, only~~ the at least one associated file **by a globally unique identifier, wherein the globally unique identifier remains constant when the at least one associated file is renamed or relocated in the server**, the selected file and the at least one associated file **further** identified by respective Uniform Resource Locators;

in response to requesting downloading of the selected file, initiating downloading of the selected file and automatically determining, at the client, the identity of and initiating downloading of the at least one associated file of the selected file to occur at substantially the same time as the selected file;

searching, by the client, for the at least one associated file using the globally unique identifier in response to a failure to find the at least one associated file in the server; and

generating respective local identifiers identifying the selected file and the at least one associated file that are indicative of the respective Uniform Resource Locators identifying the selected file and the at least one associated file;

initiating storing, in a memory associated with the client, of the selected file and the at least one associated file; and

maintaining a status file for the selected file and each of the at least one associated file.

14. (Original) The method of Claim 13, and further comprising maintaining, by the document manager, respective profiles of the one or more files.

15. (Cancelled)

16. (Previously Presented) The method of Claim 13, wherein automatically determining the identity of and initiating downloading of the at least one associated file

comprises examining a profile of the selected file, the profile identifying the at least one associated file by the Uniform Resource Locator.

17. (Original) The method of Claim 13, wherein the status file indicates one or more properties of the respective selected file and the at least one associated file.

18. (Original) The method of Claim 13, wherein the status file is a cookie file.

19. (Original) The method of Claim 13, wherein the status file consists solely of a timestamp indicative of a time of download.

20. (Original) The method of Claim 13, wherein the status file comprises a timestamp indicative of a time of download, a check out status, and respective identities of the at least one associated file.

21. (Original) The method of Claim 13, wherein the memory associated with the client is a root of a cache, the root identified by a root directory identifier.

22. (Original) The method of Claim 21, wherein each of the respective local identifiers comprises the root directory identifier.

23. (Original) The method of Claim 13, and further comprising:

generating, by the client, the one or more files for uploading to the server;

generating, by the client, a profile associated with each of the one or more files; and

uploading, by the client, the profile and the each of the one or more files to the server.

24. (**Currently Amended**) An apparatus for accessing, by a client, one or more files residing in a server comprising:

software stored on a computer readable medium and operable, when executed on a processor, to:

request downloading of a selected file residing in a server, the selected file needing access, either directly or indirectly, to at least one associated file in order to be used, the selected file including instructions to access the at least one associated file, the selected file further being associated with at least one profile **, wherein the profile identifies** ~~identifying, for download, only~~ the at least one associated file **by a globally unique identifier, wherein the globally unique identifier remains constant when the at least one associated file is renamed or relocated in the server**;

in response to the request, initiate downloading of the selected file and automatically determine, at the client, the identity of and initiate downloading of the at least one associated file of the selected file to occur at substantially the same time as the selected file; ~~and~~

search, by the client, for the at least one associated file using the globally unique identifier in response to a failure to find the at least one associated file in the server; and

initiate storing, in a memory associated with the client, of the selected file and the at least one associated file under respective local identifiers.

25. (Original) The apparatus of Claim 24, wherein each of the one or more files is associated with a profile, the profile maintained by a document manager residing in the server.

26. (Cancelled)

27. (Original) The apparatus of Claim 26, wherein the profile identifies the at least one associated file by the Uniform Resource Locator.

28. (Previously Presented) The apparatus of Claim 24, wherein the software is operable to examine a profile of the selected file in order to automatically determine the identity of and initiate downloading of the at least one associated file, the profile identifying the at least one associated file.

29. (Original) The apparatus of Claim 24, wherein the software is further operable to maintain a respective status file for each of the selected file and the at least one associated file, each status file indicating one or more properties of the respective selected file and the at least one associated file.

30. (Original) The apparatus of Claim 29, wherein the status file is a cookie file.

31. (Original) The apparatus of Claim 29, wherein the status file consists solely of a timestamp indicative of a time of download.

32. (Original) The apparatus of Claim 29, wherein the status file comprises a timestamp indicative of a time of download, a check out status, and respective identities of the at least one associated file.

33. (Original) The apparatus of Claim 24, wherein the memory associated with the client is a root of a cache, the root identified by a root directory identifier.

34. (Original) The apparatus of Claim 33, wherein each of the respective local identifiers comprises the root directory identifier.

35. (Original) The apparatus of Claim 24, wherein the software is further operable to:

generate the one or more files for uploading to the server;
generate a profile associated with each of the one or more files; and
upload the profile and the each of the one or more files to the server.

36. (Cancelled)

37. (**Currently Amended**) A system comprising:

a server having a document manager stored therein, the document manager operable to maintain a respective profile for each of a plurality of files, each profile including respective identifications of associated files associated with the file;

one or more clients associated with the server, each of the one or more clients having access to at least one computer readable medium comprising a software program operable to:

request downloading of a selected file residing in a server, the selected file needing access, either directly or indirectly, to at least one associated file in order to be used, the selected file including instructions to access the at least one associated file, the selected file further being associated with at least one profile**, wherein the profile identifies** ~~identifying, for download, only~~ the at least one associated file **by a globally unique identifier, wherein the globally unique identifier remains constant when the at least one associated file is renamed or relocated in the server**;

in response to the request, initiate downloading of the selected file and automatically determine the identity of and initiate downloading of the at least one associated file of the selected file to occur at substantially the same time as the selected file; ~~and~~

search for the at least one associated file using the globally unique identifier in response to a failure to find the at least one associated file in the server; and

initiate storing, in a memory associated with the client, of the selected file and the at least one associated file under respective local identifiers.

38. (Original) The system of Claim 37, wherein each of the identifications is the Uniform Resource Locator.

39. (Previously Presented) The system of Claim 37, wherein the software is operable to examine a profile of the selected file in order to automatically determine the identity of and initiate downloading of the at least one associated file.

40. (Original) The system of Claim 37, wherein the software is further operable to maintain a respective status file for each of the selected file and the at least one associated file, each status file indicating one or more properties of the respective selected file and the at least one associated file.

41. (Original) The system of Claim 40, wherein the status file is a cookie file.

42. (Original) The system of Claim 40, wherein the status file consists solely of a timestamp indicative of a time of download.

43. (Original) The system of Claim 40, wherein the status file comprises a timestamp indicative of a time of download, a check out status, and respective identities of the at least one associated file.

44. (Original) The system of Claim 37, wherein the memory associated with the client is a root of a cache, the root identified by a root directory identifier.

45. (Original) The system of Claim 44, wherein each of the respective local identifiers comprises the root directory identifier.

46. (Original) The system of Claim 37, wherein the software is further operable to:

generate the one or more files for uploading to the server;
generate the profile associated with each of the one or more files; and
upload the profile and the each of the one or more files to the server.

47. (Previously Presented) The method of Claim 1, further comprising in response to requesting downloading of the selected file, automatically determining the identity of and initiating downloading of at least one sub-associated file of the selected file to occur at substantially the same time as the at least one associated file, the at least one associated file needing access, either directly or indirectly, to the at least one sub-associated file in order to be used, the at least one associated file including instructions to access the at least one sub-associated file, the at least one sub-associated file identified by a separate profile associated with the at least one associated file.

48. (Previously Presented) The method of Claim 13, further comprising in response to requesting downloading of the selected file, automatically determining the identity of and initiating downloading of at least one sub-associated file of the selected file to occur at substantially the same time as the at least one associated file, the at least one associated file needing access, either directly or indirectly, to the at least one sub-associated file in order to be used, the at least one associated file including instructions to access the at least one sub-associated file, the at least one sub-associated file identified by a separate profile associated with the at least one associated file.

49. (Previously Presented) The apparatus of Claim 24, wherein the software stored on a computer readable medium is further operable, when executed on a processor, to, in response to the request, automatically determine the identity of and initiate downloading of at least one sub-associated file of the selected file to occur at substantially the same time as the at least one associated file, the at least one associated file needing access, either directly or indirectly, to the at least one sub-associated file in order to be used, the at least one associated file including instructions to access the at least one sub-associated file, the at least one sub-associated file identified by a separate profile associated with the at least one associated file.

50. (Previously Presented) The system of Claim 37, wherein the software is further operable to, in response to the request, automatically determine the identity of and initiate downloading of at least one sub-associated file of the selected file to occur at substantially the same time as the at least one associated file, the at least one associated file needing access, either directly or indirectly, to the at least one sub-associated file in order to be used, the at least one associated file including instructions to access the at least one sub-associated file, the at least one sub-associated file identified by a separate profile associated with the at least one associated file.